IN THE CLAIMS:

B7

1. (Twice Amended) A method for aligning an instruction stream comprising:

rotating data bytes of the instruction stream by a number corresponding to a byte length of a previous instruction in the instruction stream; and

shifting the data bytes to a start of the instruction immediately following the previous instruction based upon the byte length of previous the instruction.

2. (Unchanged) The method of claim 1 wherein the rotating and shifting are performed during a single clock cycle.

B8

3. (Twice Amended) The method of claim 1 further comprising:

receiving the byte length of the previous instruction from a length decode logic unit.

4. (Once Amended) The method of claim 1 further comprising:

storing instruction stream cache lines in alignment buffers prior to rotating the instruction stream.

5. (Twice Amended) The method of claim 1 wherein said shifting shifts to an exact start of the instruction immediately following the previous instruction.

6. (Unchanged) The method of claim 1 further comprising:

providing the output of the shifting step to a length decode logic unit.

7. (Twice Amended) Logic for aligning an instruction stream comprising:

a rotator logic unit for rotating data bytes of the instruction stream by a number corresponding to a byte length of a previous instruction in the instruction stream;

a shifter logic unit for shifting the data bytes to a start of the instruction immediately following the previous instruction based upon the byte length of the previous instruction.

8. (Unchanged) The logic of claim 7 wherein the rotating and shifting are performed during a single clock cycle.

9. (Twice Amended) The logic of claim 7 further comprising:

a length vector providing the byte length of the previous instruction.

10. (Once Amended) The logic of claim 7 further comprising:

alignment buffers for storing instruction stream cache lines for use by the rotator logic unit.

11. (Twice Amended) A processor to align an instruction stream comprising:

a rotator logic unit for rotating data bytes of the instruction stream by a number corresponding to a byte length of a previous instruction ;

a shifter logic unit for shifting the data bytes to a start of the instruction immediately following the previous instruction based upon the length of the previous instruction.

B10

12. (Twice Amended) The processor of claim 11 further comprising:

a length vector providing the byte length of previous instruction.

13. (Twice Amended) A system for aligning an instruction stream comprising:

means for rotating data bytes of the instruction stream by a number corresponding to a byte length of a previous instruction in the instruction stream; and

means for shifting the data bytes to a start of the instruction immediately following the previous instruction based upon the byte length of the previous instruction.

B11

14. (Once Amended) The method of claim 1 further comprising:

determining the byte length of the previous instruction.

15. (Once Amended) The method of claim 14 wherein the byte length of the previous instruction is based upon a length of an opcode and a length of immediate data.

16. (Once Amended) The method of claim 14 further comprising:

determining if an opcode extension byte is required to determine the byte length of the previous instruction.

17. (Unchanged) The method of claim 16 further comprising:

determining a memory address displacement length.

18. (Unchanged) The method of claim 17 further comprising:

determining an anticipatory length of the memory displacement for a one-byte opcode; and

determining an anticipatory length of the memory displacement for a two-byte opcode.

19. (Unchanged) The method of claim 18 further comprising multiplexing the anticipatory length for the one-byte opcode and the anticipatory length for the two-byte opcode to determine the length of the memory displacement.

IN THE DRAWINGS

Please amend the drawings as requested in the accompanying separate letter to the Draftsman.